DGT 3.0 CONNECTED VEHICLE PLATFORM

API *BANDEJA DE SALIDA* DESCRIPTION 1.2

AUTHOR: TECHNICAL DEPARTMENT (INSPIDE)

LAST MODIFICATION: 10/03/2020

Subdirección General de Gestión de la movilidad y Tecnología

Josefa Valcárcel 44

28027 Madrid

Tel: 91 301 83 85

Fax: 91 714 85 30

VERSION CONTROL

Version	Date	Author	Company	Description
1.0	16/10/2019	Technical Dpt.	INSPIDE	Creation
1.1	24/01/2020	Technical Dpt.	INSPIDE	Publication of information on incidents and works in progress
1.2	10/03/2020	Technical Dpt.	INSPIDE	Publication of the V16 signal

1 INTRODUCTION

This document contains a summary description of the *Bandeja de salida* API documentation set currently available on the platform at the following URLs:

API Name	URL
Bandeja de salida	{{URLservicioBandejaSalida}} /swagger-ui.html

This includes the MockServer. These are open source simulation frameworks for HTTP and HTTPS. It is designed to simplify integration testing by simulating HTTP and HTTPS as a web service or website, and to decouple development teams by allowing a team to develop against a service that is not complete or unstable.

The following describes the summary content that includes explanatory text for current or potential users.

2 BANDEJA DE SALIDA DGT 3.0

INTRODUCTION

The DGT 3.0 Connected Vehicle Platform is a platform that offers road safety and intelligent mobility services under the SaaS concept based on spatial information processing.

The services exposed in DGT 3.0 cover all the phases of the life cycle of a spatial data:

- Real-time processing of large volumes of spatial and alphanumeric data for consumption.
- Spatial analysis and application of different types of road safety logic on the Big Data available for obtaining KPIs.
- Sending of road safety information when the risk level exceeds a preset threshold.

The services offered by DGT 3.0 contribute to the improvement of road safety and mobility through the prevention of traffic accidents and the minimization of their effects, especially on people's lives and health.

The **DGT 3.0 Broadcast Tray API [BD DGT 3.0]**, whose interfaces and use are described on this page, allows Users (drivers, pedestrians, cyclists, emergency services, roadside assistance services, road maintenance services, etc.) to receive geo-referenced notifications if their risk level so advises.

This risk level can be evaluated according to different parameters such as *speed*, *mode of transport*, *status*, *weather*, *road condition*, *proximity of events*, *elements or users*, *etc*. and is assigned to each connected device.

If the road risk level of a device exceeds a certain threshold, it is sent information on the event that increases its risk so that, applying the user information policy that it considers appropriate, it can notify the device by the means that it considers most appropriate, in order to prevent the occurrence of road accidents.

The **DGT 3.0 *Bandeja de Salida* API [BS DGT 3.0]**, whose interfaces and use are described in this page, allows Users (Companies, institutions, groups belonging to the DGT 3.0 ecosystem) to request on demand **information about georeferenced events** that impact user mobility.

Since version 1.1 of this document, information about the events recorded in the LINCE application, as well as Works in Progress, is published. For its publication, a mapping has been made between LINCE events and VMP events, so that the first ones have been categorized within the second ones, assigning them the corresponding icon. The text published is the initial of the LINCE event.

In the case of works, these are published as category **4 - Works**, from the **category** table, having the corresponding icon.

API BS DGT 3.0

This document describes the interfaces and use of the BS DGT 3.0 API of the Road Safety and Mobility Platform "**DGT 3.0 Connected Vehicle Platform**" offered by DGT3.0.

The API BS DGT 3.0 allows the sending of data to the Platform, to which it responds with the information of events related to road safety according to the attributes requested by the User at any time.

This is the first version of the API BS DGT 3.0 and it is open to the use of the developers of mobile applications, manufacturers or others who wish to use it according to the Terms of Use established by the *Dirección General de Tráfico* within the framework of the project DGT 3.0.

> **The Directorate General of Traffic wants developers, entities and companies to join the Collaborative Road Safety and build together a new form of mobility! Let's make our streets and roads a safer place. The potential is huge, join us!**

ENVIROMENTS

An environment has been deployed to allow developers to perform the necessary tests. An access token must be requested for each company.

TABLAS DE REFERENCIA

***category**:* Event category

Label of the Type of Event indicated by the VMP.

category	Description
1	Unknow / general
2	Vulnerable
3	Incidence
4	Work
5	Conservation
6	State of infrastructure
7	Boarding Times
8	Accident / Detainee
9	Weather
10	Traffic conditions
11	Radar
12	Obstacle
13	Event
14	Restriction
15	Air quality
16	Ports
17	Shedding
18	Estado parking
19	Fixed obstacle
20	Mobile obstacle
21	Regulatory measures
22	Bagging
23	Capacity alteration
24	Road alteration

***mode**:* Transport mode

Types of transport modes covered by the platform and which are used in the VMP to indicate which transport mode is affected by the VMP.

mode	Description
3	Moped / Motorcycle
4	Animal traction
5	Tractor
6	Motorhome
7	Tourism
8	Van/Truck
9	Bus or coach
10	Truck
11	All

***direction**:* Direction

Attribute indicating the direction of traffic to which the standard VMP applies (point or section)

direction	Description
1	Both
2	Growing
3	Decreasing

***type**:* VMP type

type	Description
1	Point
2	Section
3	Area
99	All

***withgeom**:* Geometry

Return of VMP geometry is required or not in the response.

direction	Description
1	Yes
2	No

***event**:* Event

Attribute that indicates the form of consumption of the information that applies to the area type VMP.

direction	Description
1	At the entrance
2	At the exit

***errorCode**:* Error codes

Types of transport modes covered by the platform and which are used in the VMP to indicate which transport mode is affected by the VMP.

modo	Description
1	idCompany and category fields must be sent
2	province must be sent when road is requested
3	province must be an integer value
4	type must be an integer value
5	kp from must be lower than kpto
6	If type 3 is defined, eventarea field must be sent
7	direction must be an integer value
8	If type 3 is defined, eventarea field must be sen
99	General error

Incidents reported through signal V16

The information generated by the devices that provide the V16 signal is integrated into the DGT3.0 platform and published through the *bandeja de salida*. Through it, it appears with the text "V16 event" and with the icon P50O.png, as shown in the image.

The operation of the input of this signal is described in the document "Definition of protocols and interfaces for the V16 signal v0.3.pdf

Incidents LINCE and Works: The set of incidents related to the information published by LINCE regarding incidents and works is described in detail in the document 20191010_DGT30_1.0.xlxs The fields provided are the following. Their content is not published, since they are internal data to the application that are used for the mapping between LINCE and VMP events.

Modo	Description
IDS_UC	Event identifier
SUC_LINCE_1	Event identifier 1 on LINCE
SUC_LINCE_2	Event identifier 2 on LINCE
ID_CAT	Category identifier in VMP
ID_ICO	LINCE event icon identifier in VMP

REFERENCES

***getProvinces**:* Obtaining literals of Provinces and their internal identification codes.

+Response 200 (application/json)


```
{
   "errorCode": 0,
   "errorDesc": "OK",
   "data": [
      {
         "provinceId": 1,
         "provinceName": "Álava"
      },
      {
         "provinceId": 2,
         "provinceName":"Albacete"
      }
   ]
```


***getRoads**:* Obtaining of literals of Roads and the internal codes of identification of the Provinces.

+Response 200 (application/json)


```
{
   "errorCode": 0,
   "errorDesc":
   "OK", "data": [
      {
         "provinceId": 12,
         "roadsName": "A-226"
      },
      {
         "provinceId": 12, "roadsName": "A-23"

      }
   ]
}
```


***getRoadsDirection**:* Obtaining Address Literals and their Internal Identification Codes.

+Response 200 (application/json)


```
{
   "errorCode": 0,
   "errorDesc": "OK",
   "data": [
      {
         "directionId": 1,
         "directionName": "Ambos"
      },
      {
         "directionId": 2,
         "directionName": "Creciente"
      },
      {
         "directionId": 3,
         "directionName": "Decreciente"
      }
   ]
}
```

getModes: *Obtaining of literals of Modes of Transport and their internal identification codes.*

+Response 200 (application/json)


```
{
   "errorCode": 0, "errorDesc": "OK",
   "data": [
      {
       "modeId": 9,
       "modeName": "Autobús o autocar"
    },
    {
       "modeId": 6,
       "modeName":"Autocaravana"
    },
    {
       "modeId": 1,
        "modeName": "Bicicleta"
    },
    {
       "modeId": 10,
       "modeName": "Camión"
    },
    {
       "modeId": 3,
       "modeName": "Ciclomotor / Motocicleta"
    },
    {
       "modeId": 8,
       "modeName":"Furgón/furgoneta"
    },
    {
       "modeId": 2,
       "modeName": "Peatón"
    },
    {
       "modeId": 11,
       "modeName": "_Todos_"
    },
    {
       "modeId": 4,
       "modeName": "Tracción animal"
    },
    {
       "modeId": 5,
       "modeName": "Tractor"
      },
      {
         "modeId": 7,
         "modeName": "Turismo"
      }
   ]
}
```


getCategories: Obtaining literals of event categories and their internal identification codes.

+ Response 200 (application/json)


```
{
   "errorCode": 0,
   "errorDesc": "OK",
   "data": [
      {
```

```
      "categoryId": 8,
      "categoryName": "Accidente / Detenido"
    },
    {
      "categoryId": 23,
      "categoryName": "Alteración capacidad"
    },
    {
      "categoryId": 24,
      "categoryName": "Alteración carretera"
    },
    {
      "categoryId": 15,
      "categoryName": "Calidad del aire"
    },
    {
      "categoryId": 5,
    "categoryName":"Conservación"
  },
  {
    "categoryId": 17,
    "categoryName":"Derramamiento"
  },
  {
    "categoryId": 1,
    "categoryName": "Desconocido / general"
  },
  {
    "categoryId": 22,
    "categoryName":"Embolsamiento"
  },
  {
    "categoryId": 6,
    "categoryName": "Estado de la infraestructura"
  },
  {
    "categoryId": 10,
    "categoryName": "Estado del tráfico"
  },
  {
    "categoryId": 18,
    "categoryName": "Estado parking"
  },
  {
    "categoryId": 13,
    "categoryName":"Eventos"
  },
  {
    "categoryId": 3,
    "categoryName":"Incidencia"
  },
  {
    "categoryId": 21,
    "categoryName": "Medidas de regulación"
  },
  {
    "categoryId": 9,
    "categoryName":"Meteorológica"
  },
  {
    "categoryId": 4,
    "categoryName": "Obra"
  },
  {
    "categoryId": 12,
    "categoryName":"Obstáculo"
  },
  {
```

```
            "categoryId": 19,
            "categoryName": "Obstáculo fijo"
        },
        {
            "categoryId": 20,
            "categoryName": "Obstáculo móvil"
        },
        {
            "categoryId": 16,
            "categoryName": "Puertos"
        },
        {
            "categoryId": 11,
            "categoryName": "Radar"
        },
        {
            "categoryId": 14,
            "categoryName": "Restricción"
        },
        {
            "categoryId": 7,

            "categoryName": "Tiempos de embarque"
        },
        {
            "categoryId": 2,
            "categoryName": "Vulnerable"
        }
    ]
}
```


getTypes: Obtaining literals of VMP types and their internal identification codes.

+ Response 200 (application/json)


```
{
    "errorCode": 0,
    "errorDesc": "OK",
    "data": [
        {
            "type": 1,
            "typeName": "Punto"
        },
        {
            " type ": 2,
            " typeName ": "Tramo"
        },
        {
            " type ": 3,
            " typeName ": "Area"
        }
    ]
}
```


getEventsArea: Obtaining the literals of the errors and their internal identification codes.

+ Response 200 (application/json)


```
    {
        "errorCode": 0,
        "errorDesc": "OK",
```

```
    "data": [
      {
        "eventId": 1,
        "eventName": "Al Entrar"
      },
      {
        "eventId": 2,
        "eventName": "Al Salir"
      }
    ]
  }
```


getErrorCodes: Obtaining the literals of the errors and their internal identification codes.

+ Response 200 (application/json)


```
  {
    "errorCode": 0,
    "errorDesc": "OK",
    "data": [
      {
        "errorCode": 1,
        "errorDesc": "idCompany and category fields must be sent"
      },
      {
        "errorCode": 2,
        "errorDesc": "province must be sent when road is requested"
      },
      {
        "errorCode": 3,
        "errorDesc": "province must be an integer value"
      },
      {
        "errorCode": 4,
        "errorDesc": "type must be an integer value"
      },
      {
        "errorCode": 5,
        "errorDesc": "kp from must be lower than kpto"
      },
      {
        "errorCode": 6,
        "errorDesc": "If type 3 is defined, eventarea field must be sent"
      },
      {
        "errorCode": 7,
        "errorDesc": "direction must be an integer value"
      },
      {
        "errorCode": 8,
        "errorDesc": "If type 3 is defined, eventarea field must be sent"
      },
      {
        "errorCode": 99,  "errorDesc": "General error"
      }
    ]
  }
```

getPmvvss: Obtaining notifications by risk levels.

REQUEST

ATTRIBUTES	VALUES	DESCRIPTION
idcompany *required*	-	string common name contained in the client certificate
token *required*	-	string API key corresponding to the client platform. Obtained through the getToken operation
type *optional*	1 a 3, 99	number Type of VMP point, leg or area, obtained from getTypes
province *optional*	1 a 50	number Province identification code, obtained from getProvinces
road *optional*	-	string Road identification code, obtained from getRoads taking into account the 'province' attribute
kpfrom *optional*	-	number Initial mileage point
Kpto *optional*	-	number Final mileage point
direction *optional*	1 a 3	number Road Address Identification Code, obtained from getRoadsDirection
mode *optional*	1 a 10	number Type of Transport Mode, obtained from getModes
category *optional*	0 a 23	number Type of Event Category, obtained from getCategories
withgeom *optional*	1,2	number Request for inclusion or not of geometries, obtained from getCategories

Example A

Attribute	Attribute	Value	Literally
VMP Type	type	0	Point
Province	province	28	Madrid
Carretera	road	A-2	*na*
Initial mileage point	kpfrom	2	*na*
Final mileage point	kpto	50	*na*

Attribute	Attribute	Value	Literally
Direction	direction	3	Both
Transport Mode	mode	7	Tourism
Category	category	7	Accident / Detainee
Geometry	withgeom	1	yes

REQUEST: Example A


```
{
   "idcompany": "its.ctag.es",
   "token":      "7d001186aa0fe99b868474b355cc446b0ce99bbc2b77ab4c4cb100e1dfae152e",    "type":1,
   "province": 36,  "road": "N-550",
   "kpfrom": 2,
   "kpto": 50,
   "direction": 2,
   "mode": 7,
   "category": 7,
   "withgeom": 1
}
```


RESPONSE: Example A


```
{
   "errorCode": 0,
   "errorDesc": "OK",
   "data": [
      {
         "pmvGeomWkt": "POINT(-8.3561 43.2397)",
         "pmvId": 393,
         "pmvMsg": "Accidente múltiple",
         "pmvType": 1,
         "pmvProv": 36,
         "pmvRoad": "N-550",
         "pmvPk": 19,
         "pmvPkIni": 0,
         "pmvPkFin": 0,
         "pmvDirection": 2,
         "pmvCategory": 7,
         "pmvMode": "3,5,6,7,8,9,10",
         "pmvProvFin": 0,
         "pmvRoadFin": null,
         "pmvEvent": 0
      },
      {
         "pmvGeomWkt": "POINT(-8.3581 43.2469)",
         "pmvId": 394,
         "pmvMsg": "Vehículo detenido en arcén",
         "pmvType": 1,
         "pmvProv": 36,
         "pmvRoad": "N-550",
         "pmvPk": 18,
         "pmvPkIni": 0,
         "pmvPkFin": 0,
         "pmvDirection": 2,
         "pmvCategory": 7,
         "pmvMode": "3,5,6,7,8,9,10",
```

```
            "pmvProvFin": 0,
            "pmvRoadFin": null,
            "pmvEvent": 0
        }
    ]
}
```


ATTRIBUTES	DESCRIPTION
errorCode *required*	**number** Error code. Value among those available in the getErrorCodes operation: 1 to 8, 99. *0*
errorDesc *required*	**string** Description of the error. Text value describing the error. 1 a 8, 99. *OK*
data *required*	**array** **ARRAY (Object)**

ATRIBUTOS	VALORES	DESCRIPCIÓN
pmvGeomWkt *required*	Geometry in format WKT	**string** VMP Geometry. *POINT(-8.3561 43.2397)*
pmvId *required*	-	**number** VMP identifier. *393*
pmvMsg required	-	**string** VMP text message. *Multiple Accident*
pmvType *required*	1 a 3, 99	**Number** VMP Type.This is one of the values obtained through the getTypesPmv operation. *1*
pmvProv *required*	1 a 50	**number** Province / Initial Province. *36*
pmvRoad *required*	-	**String** Road / Initial road. *N-550*
pmvPk *required*	-	**number** Kilometer point.

ATTRIBUTES	DESCRIPTION		
			19
	pmvPkIni *required*	-	**number** Initial mileage point. *0*
	pmvPkFin *required*	-	**number** Final mileage point *0*
	pmvDirection *required*	1 a 3	**Number** Information consumption direction in Point / Stretch type VMPs. *2*
	pmvCategory *required*	1 a 24	**number** VMP Category. 7
	pmvMode *required*	1 a 11	**String** Modes of transport affected. *3,5,6,7,8,9,10*
	pmvProvFin *required*	1 a 50	**String** Final province. 0
	pmvRoadFin *required*	-	**string** Road / Final road. *null*
	pmvEvent *required*	1 a 2	**number** Form of information consumption in VMPs. *0*

Response: 500

BODY


```
{
"errorCode": 1,
"errorDesc": "identificador no válido"
}
```